UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/646,787	08/25/2003	Craig Hansen	43876-145	3618

7590 05/09/2005

McDermott, Will & Emery
600 13th Street, N.W.
Washington, DC 20005-3096

EXAMINER	
TSAI, HENRY	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 05/09/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/646,787	HANSEN ET AL.	
	Examiner	Art Unit	
	Henry W.H. Tsai	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *23 December 2003*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-30* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *12/23/03*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Specification

1. The disclosure is objected to because of the following informalities:

at page 21, line 4, "1310" should read -1210-.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

2. Claims 1-30 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

In claim 1, lines 14-16, it is not clear what is meant by "the multi-precision execution unit configurable to dynamically partition data received from the data path to account for an elemental width of the data" since it is not understandable. How can a dynamically partitioning account for an elemental width of the data ? Besides, what is really meant by "account for an elemental width of the data" is unclear.

In claim 1, line 19, it is not clear whether the "operand registers" is the same as or inside the "at least one register file" mentioned in line 12. If so, it is suggested to be clearly indicated in the claim.

Applicant is required to review the claims and correct all language which does not comply with 35 U.S.C. § 112, second paragraph.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless -

> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 1-30 are, as best understood, rejected under 35 U.S.C. 102(e) as being anticipated by Agarwal et al. (U.S. Patent No. 5,887,183) herein referred to as Agarwal et al.'183.

Referring to claim 1, Agarwal et al.'183 discloses as claimed a data processing system (see Fig. 2) comprising: (a) a bus (data bus 126, instruction bus 106 or address bus 108 see Fig. 2) coupling components (such as elements 102, 104, 112, and 116 see Fig. 2) in the data processing system; (b) an external memory (104, see fig. 2) coupled to the bus; (c) a programmable microprocessor (100, see Fig. 2) coupled to the bus and capable of operation independent of another host processor (since superscalar processor 100 as shown in Fig. 2 is an independent system), the microprocessor comprising: a virtual memory addressing unit (branch execution unit 102, see Fig. 2, see also col. 7, lines 18-19 regarding branch execution unit 102 providing address signals to memory 104); an instruction path (118 see Fig. 2) and a data path (such as 122, 124, or 126 see Fig. 2); an external interface (interface unit 216 see Fig. 2) operable to receive data from an external source (memory 104 see Fig. 2) and communicate the received data over the data path (126 see Fig. 2); a cache (such as a portion of vector registers 238 in PE0 230, 232, or 234 see Fig. 2) operable to

retain data communicated between the external interface and the data path; at least one register file (such as register file 236 in PE1 232 see Fig. 2) configurable to receive and store data from the data path and to communicate the stored data to the data path; and a multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) coupled to the data path, the multi-precision execution unit configurable to dynamically partition data received from the data path (note the data VR0 inside vector registers 238 is a partitioned data see Fig. 4A) to account for an elemental width of the data wherein the elemental width of the data is equal to or narrower than the data path (the width of data element 0 inside VR0 is equal to or narrower the data path 136 as shown in Fig. 4A), the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) being capable of performing group floating-point operations (since there are a plurality of FPUs 244 in PE0 - PE n-1 see Fig. 2) on multiple operands in partitioned fields of operand registers and returning catenated results (since a vector operation is processed by the Agarwal et al.'183's system, see also col. 6, lines 49-51).

As to claim 2, Agarwal et al.'183 also discloses: the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) is capable of performing group add, group subtract and group

multiply arithmetic operations (by Floating-Point execution unit 112 and a plurality of FPUs 244 in PE0 - PE n-1 see Fig. 2) on catenated floating-point data and, for each such group operation, returning catenated results of the operation to a register (such as vector registers 238 see Fig. 3 or FPRs of Floating-Point execution unit 112 see Fig. 2).

As to claim 3, Agarwal et al.'183 also discloses: at least some of the group add, group subtract and group multiply arithmetic operations perform arithmetic operations on floating-point data stored in first and second operand registers (such as vector registers 238 see Fig. 3) and return the catenated result to a result register (inside vector registers 238 see Fig. 3).

As to claim 4, Agarwal et al.'183 also discloses: the result register is a different register than either the first or second operand registers (note inside vector registers 238, there are many register files 236, some being used as the first or second operand registers and another one being used as a result register see Fig. 3).

As to claims 5 and 24, Agarwal et al.'183 also discloses: the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) is capable of executing a first plurality of group floating-

point operations on floating-point data of a first precision (single precision data 424, see Col. 18, line 4) and a second plurality of group floating-point operations on floating-point data of a second precision (double precision data, see Col. 18, line 10) that is a higher precision than the first precision and wherein a number of data elements stored in partitioned fields of the operand registers (such as register file 236 in PE1 232 see Fig. 3) for the first and second plurality of group floating-point operations is inversely (since double precision data (64-bit element) need more register space than the single precision data (32-bit element)) related to the precision of the data elements.

As to claim 6, Agarwal et al.'183 also discloses: the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) is capable of executing group floating point operations that operate on 32-bit data elements (single precision data 424 (32-bit element), see Col. 18, lines 4, 43 and 44) and group floating-point operations that operate on 64-bit data elements (double precision data (64-bit element), see Col. 18, lines 10, 43, and 44) and wherein a number of data elements stored in partitioned fields of operand registers (such as register file 236 in PE1 232 see Fig. 2) used for the operations that operate

on 32-bit data elements is twice as many as a number of data elements stored in partitioned fields of operand registers (register file 236 in PE1 232 see Fig. 3) used for the operations that operate on 64-bit data elements.

As to claims 7 and 25, Agarwal et al.'183 also discloses: when performing at least some of the group floating-point operations, the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) operates on partitioned fields of operand registers in parallel (since there are a plurality of FPUs 244 in PE0 - PE n-1 see Fig. 2) and returns the catenated results to a register (since a vector operation is processed by the Agarwal et al.'183's system, see also col. 6, lines 49-51).

As to claims 8 and 26, Agarwal et al.'183 also discloses: the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) is capable of executing a plurality of group floating-point operations (since there are a plurality of FPUs 244 in PE0 - PE n-1 see Fig. 2) on floating-point data of a first precision (single precision data 424 (32-bit element), see Col. 18, lines 4, 43 and 44) and a plurality of group floating-point operations on floating-point data of a second precision (double precision data (64-bit element), see Col. 18, lines 10, 43, and 44) that is a higher precision than the first precision and wherein, when

performing at least one of the group floating-point operations on floating-point data of the first precision, the multi-precision execution unit operates on at least two partitioned operands in parallel (since there are a plurality of FPUs 244 in PE0 - PE n-1 see Fig. 2).

As to claim 9, Agarwal et al.'183 also discloses: the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) is capable of performing group floating-point operations (since there are a plurality of FPUs 244 in PE0 - PE n-1 see Fig. 2) on catenated data having a total aggregate width of 128 bits (this is the situation when 3 32-bit data or two 64-bit data are operated by the Agarwal et al.'183's system).

As to claim 10, Agarwal et al.'183 also discloses: the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) is capable of performing group floating-point operations on floating-point data of more than one precision (since as set forth above there are 32-bit single precision and 64-bit double precision data used by the Agarwal et al.'183's system).

As to claim 11, Agarwal et al.'183 also discloses: the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) is capable of performing group integer operations (this is the

situation when fixed point execution units FXUs 242 are used to process the data, see Col. 9, lines 42 and Fig. 3) on multiple operands in partitioned fields of operand registers and returning catenated results to a register (since a vector operation is processed by the Agarwal et al.'183's system, see also col. 6, lines 49-51).

As to claim 12, Agarwal et al.'183 also discloses: the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) is capable of performing group add, group subtract and group multiply arithmetic operations (this is the situation when fixed point execution units FXUs 242 are used to process the data, see Col. 9, lines 42 and Fig. 3) on catenated integer data and, for each such group operation, returning catenated results of the operation to a register (since a vector operation is processed by the Agarwal et al.'183's system, see also col. 6, lines 49-51).

As to claim 13, Agarwal et al.'183 also discloses: at least some of the group add, group subtract and group multiply arithmetic operations (as set forth above this is the situation when fixed point execution units FXUs 242 are used to process the data, see Col. 9, lines 42 and Fig. 3) perform arithmetic operations on integer data stored in first and second operand

registers and return the catenated result to a result register (since a vector operation is processed by the Agarwal et al.'183's system, see also col. 6, lines 49-51).

As to claim 14, Agarwal et al.'183 also discloses: the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) is capable of executing a first plurality of group integer operations on integer data (as set forth above this is the situation when fixed point execution units FXUs 242 are used to process the data, see Col. 9, lines 42 and Fig. 3) of a first precision (single precision data 424, see Col. 18, line 4) and a second plurality of group integer operations on integer data of a second precision (double precision data, see Col. 18, line 10) that is a higher precision than the first precision and wherein a number of data elements stored in partitioned fields of the operand registers (such as register file 236 in PE1 232 see Fig. 3) for the first and second plurality of group integer operations is inversely (since double precision data (64-bit element) need more register space than the single precision data (32-bit element)) related to the precision of the data elements.

As to claim 15, Agarwal et al.'183 also discloses: when performing at least some of the group integer operations (as set forth above this is the situation when fixed point execution

units FXUs 242 are used to process the data, see Col. 9, lines 42 and Fig. 3), the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) operates on partitioned fields of operand registers in parallel (since there are a plurality of FXUs 244 in PE0 – PE n-1 see Fig. 2) and returns the catenated results to a register (since a vector operation is processed by the Agarwal et al.'183's system, see also col. 6, lines 49-51).

As to claim 16, Agarwal et al.'183 also discloses: the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) is capable of executing a plurality of group integer operations (as set forth above this is the situation when fixed point execution units FXUs 242 are used to process the data, see Col. 9, lines 42 and Fig. 3) on integer data of a first precision (single precision data 424, see Col. 18, line 4) and a plurality of group integer operations on integer data of a second precision (double precision data, see Col. 18, line 10) that is a higher precision than the first precision and wherein, when performing at least one of the group integer operations on integer data of the first precision, the multi-precision execution unit operates on at least two partitioned operands (such as register file 236 in PE1 232 see Fig. 3) in parallel (since there are a plurality of FXUs 244 in PE0 – PE n-1 see Fig. 2).

As to claim 17, Agarwal et al.'183 also discloses: the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) is capable of performing one or more group data handling operations that operate on multiple operands in partitioned fields of one or more operand registers (such as register file 236 in PE1 232 see Fig. 3) and returning catenated results to a register (since a vector operation is processed by the Agarwal et al.'183's system, see also col. 6, lines 49-51).

As to claim 18, Agarwal et al.'183 also discloses: the one or more group data handling operations comprises a first group operation that converts a plurality of n-bit (64-bit, see col. 18, lines 6) data elements in a first operand register and a plurality of n-bit data elements in a second operand register into a plurality of n/2-bit (32-bit, see col. 18, lines 6) data elements in a result register (see Col. 18, lines 4-7 regarding obtaining 32-bit data from 64 bit elements).

As to claim 19, Agarwal et al.'183 also discloses: the first group operation shifts each of the plurality of n/2-bit data elements by a specified number of bits during the conversion (note certainly the conversion of the data involves shifting bits in the Agarwal et al.'183's system).

As to claim 20, Agarwal et al.'183 also discloses: the one or more group data handling operations comprises a second group operation that interleaves a plurality of data elements selected from a first operand register (such as register file 236 in PE0 232 see Fig. 3) with a plurality of data elements selected from a second operand register (such as register file 236 in PE1 232 see Fig. 3) and catenates the data elements into a result register.

As to claim 21, Agarwal et al.'183 also discloses: the one or more data handling operations comprises a group shift left operation that shifts bits of individual data elements catenated in an operand register to the left and clears empty low order bits of the individual data elements to zero (note this is the situation when the value of operand is to be multiplied by 2, one left shift one time is certainly operated in the Agarwal et al.'183's system).

As to claim 22, Agarwal et al.'183 also discloses: the one or more data handling operations comprises a group shift right operation that shifts bits of individual data elements catenated in an operand register to the right and fills empty high order bits of the individual data elements with a value equal to a value stored in a sign bit of the individual data element (note

this is the situation when the value of operand is to be divided by 2, one right shift one time is certainly operated in the Agarwal et al.'183's system and filling empty high order bits of the individual data elements with a value equal to a value stored in a sign bit in order to maintain the same sign value is a certain step in the Agarwal et al.'183's system when 2's complement system is used).

As to claim 23, Agarwal et al.'183 also discloses: the one or more data handling operations comprises a group shift right operation that shifts bits of individual data elements catenated in an operand register to the right and clears empty high order bits of the individual data elements to zero (note this is the situation when the value of operand is to be divided by 2, one right shift one time is certainly operated in the Agarwal et al.'183's system and clearing empty high order bits of the individual data elements to zero is a certain step in the Agarwal et al.'183's system when unsigned system is used).

As to claim 27, Agarwal et al.'183 also discloses: the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) comprises a plurality of functional units (PE0-PEn-1 see Fig. 2).

As to claim 28, Agarwal et al.'183 also discloses: the at least one register file (such as register file 236 in PE1 232 see Fig. 3) comprises a plurality of registers (register 238 see Fig. 3) that can be used to store operands and results for the group floating-point operations.

As to claim 29, Agarwal et al.'183 also discloses: the multi-precision execution unit (such as PE0-PEn-1 see Fig. 2) returns the catenated results to a register (since a vector operation is processed by the Agarwal et al.'183's system, see also col. 6, lines 49-51).

As to claim 30, Agarwal et al.'183 also discloses: the at least one register file (such as register file 236 in PE1 232 see Fig. 3) comprises a plurality of general purpose registers (register 238 see Fig. 3) that can be used as operand and result registers for group floating-point operations (since there are a plurality of FPUs 244 in PE0 - PE n-1 see Fig. 2).

Conclusion

4. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Ashburn et al.'340 discloses a reduced area floating point processor control logic utilizing a decoder between a control unit and the FPU. the floating point processor decode the control signals to provide the FPU signals. If the number of control signals is one less than the number of FPU signals, a priority encoder is used as the decoder, unless the FPU signals include a power savings signal. Otherwise a custom decoder is used.

Cocke et al.'938 discloses an instruction control mechanism for a computing system with register renaming, map table and queues indicating available registers. A floating point instruction control mechanism which processes loads and stores in parallel with arithmetic instructions.

Contact Information

5. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Dr. Henry Tsai whose telephone number is (571) 272-4176. The examiner can normally be reached on Monday-Thursday from 8:00 AM to 5:00 PM. If attempts to reach the examiner by telephone are unsuccessful, the examiner supervisor, Eddie Chan, can be reached on (571) 272-4162. Any inquiry of a general nature or relating to the

status of this application or proceeding should be directed to **the TC central telephone number, 571-272-2100.**

6. In order to reduce pendency and avoid potential delays, Group 2100 is encouraging FAXing of responses to Office actions directly into **the Group at fax number: 703-872-9306.** This practice may be used for filing papers not requiring a fee. It may also be used for filing papers which require a fee by applicants who authorize charges to a PTO deposit account. Please identify the examiner and art unit at the top of your cover sheet. Papers submitted via FAX into Group 2100 will be promptly forward to the examiner.

HENRY W. H. TSAI
PRIMARY EXAMINER

May 5, 2005